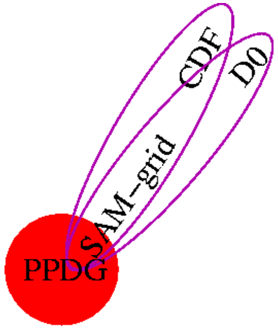
# Distributed Computing at CDF



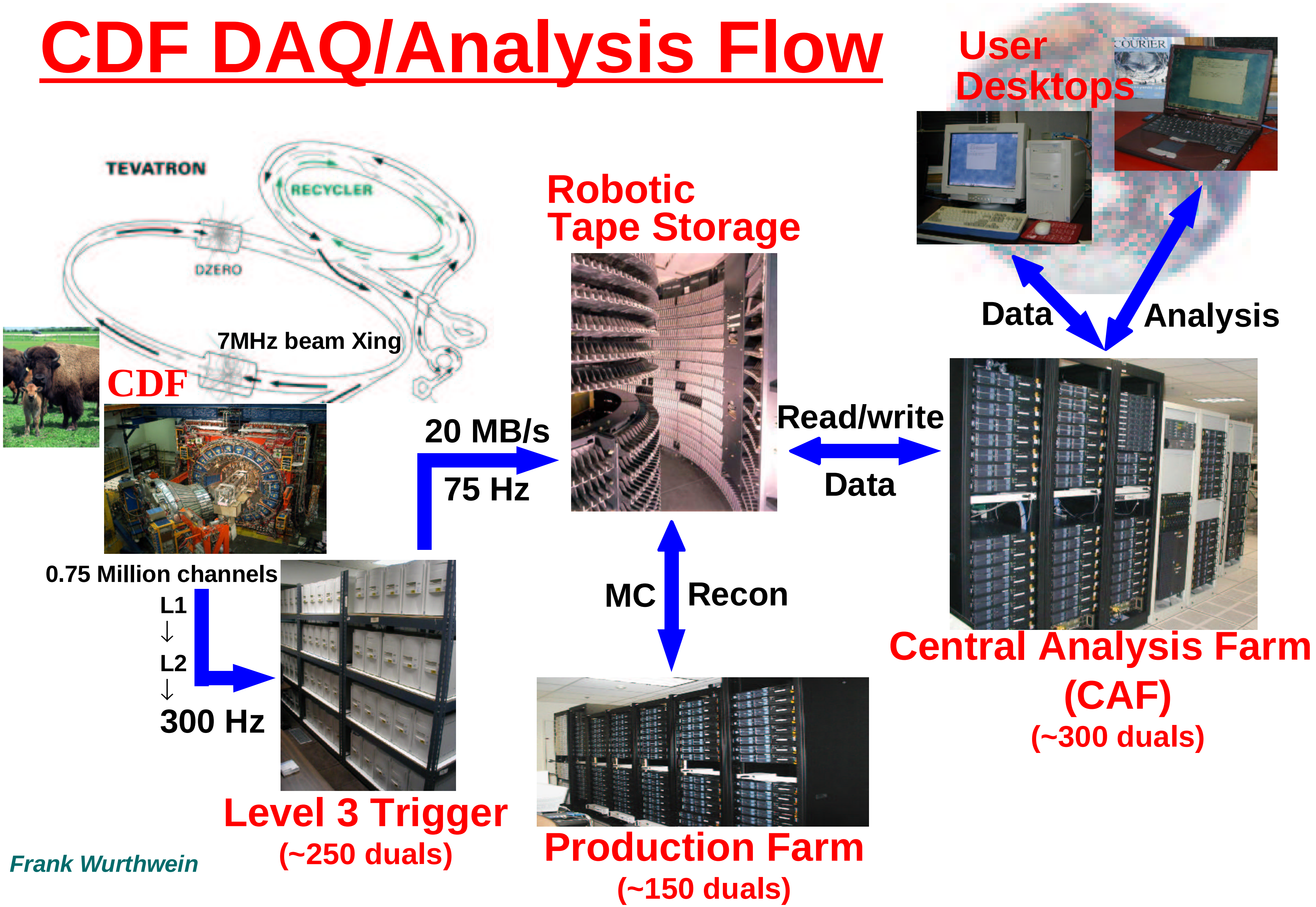
**Frank Wurthwein**

*MIT/UCSD/FNAL-CD*

for the CDF Collaboration

- Computing Model
- CDF Today
- PPDG activities today
- Future directions

# CDF DAQ/Analysis Flow



7MHz beam Xing

CDF

0.75 Million channels

L1
↓
L2
↓
300 Hz

Level 3 Trigger
(~250 duals)

20 MB/s
75 Hz

Robotic Tape Storage

MC
Recon

Production Farm
(~150 duals)

Read/write
Data

Central Analysis Farm (CAF)
(~300 duals)

User Desktops

Data
Analysis

*Frank Wurthwein*

# Data/Software Characteristics

## Data Characteristics:

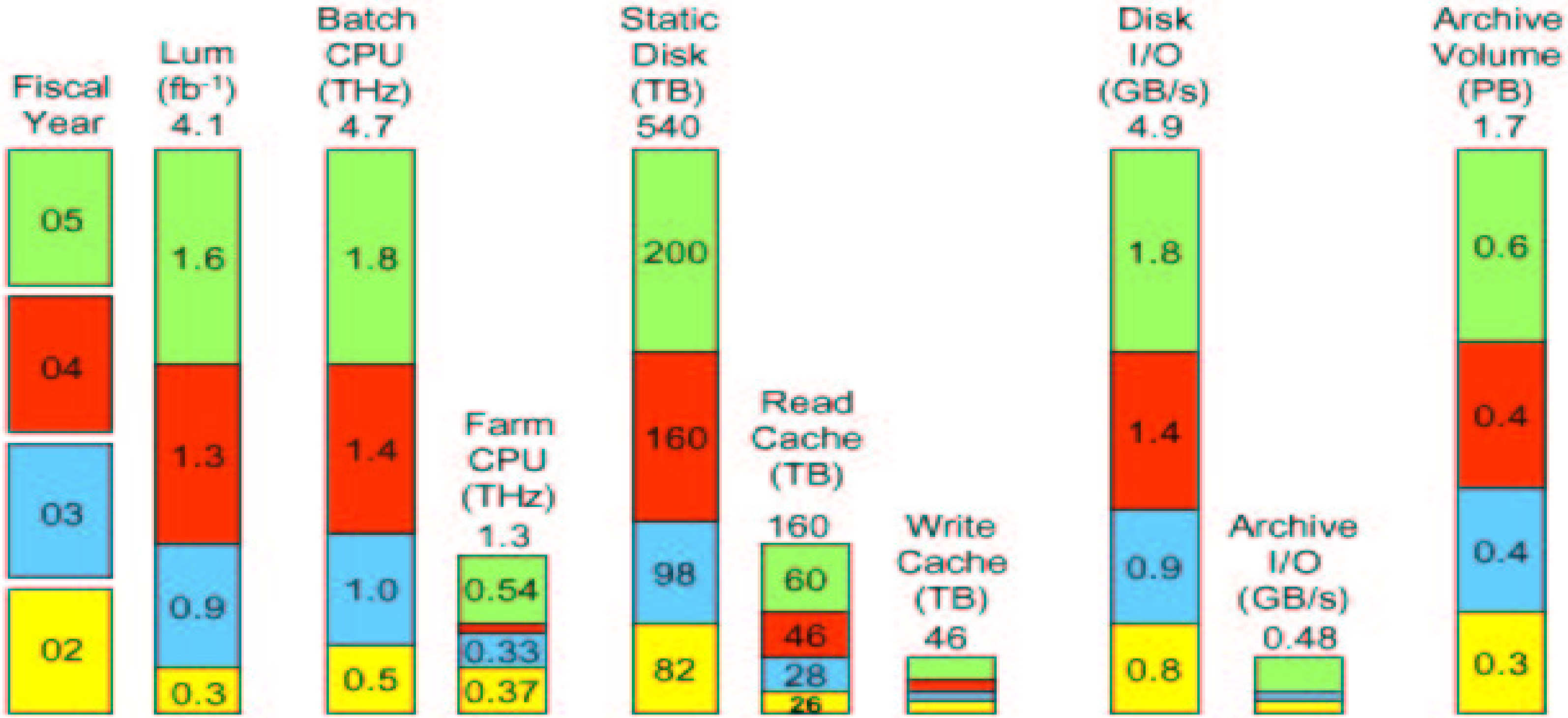
- Root I/O: ~80-400 kB/event (configurable content)
- 'Standard' ntuple: 5-10 kB/event
- Typical RunIIa secondary dataset size: $10^7$ events
- Winter03 physics: ~100 datasets adding up to ~50TB
- Largest dataset for Winter03 physics: 3.5e7 evts
- Expect twice the data for Summer03

## Analysis Software:

- Typical analysis jobs run @ few Hz on 1 GHz P3

  → few MB/sec
- CPU rather than I/O bound (FastEthernet)

# Computing Requirements



**Requirements set by goal:**

200 simultaneous users to analyze secondary data set ($10^7$ evts) in a day

**Need ~700 TB of disk and ~5 THz of CPU by end of FY'05:**

**2 Million $$$ hardware budget/year**

# Computing Model

## Interactive Computing on desktop:

- Complete access to all data from desktop via dCache & rootd

## Batch Computing on "remote" cluster(s):

- Binary compatible with desktop
- qsub, qstat, kill, ls, tail, top via command line/web
- Large scale parallelisation with single submission
  - Single summary email upon completion
- User scratch space inside cluster
  - Krb5 ticket created @ launch time
- Data access Winter03: 90% NFS+rootd, 10% dCache
- Summer03: 70% dCache, 30% NFS+rootd

# User Analysis Today

## Deployed Hardware @ FNAL:

- ~180TB disk space, ~300TB data on tape
- 600 user analysis CPUs (=1THz)
- 100's of desktops & 2 central 8-ways & legacy smp

& infrastructure HW like code servers, DB, www, ...

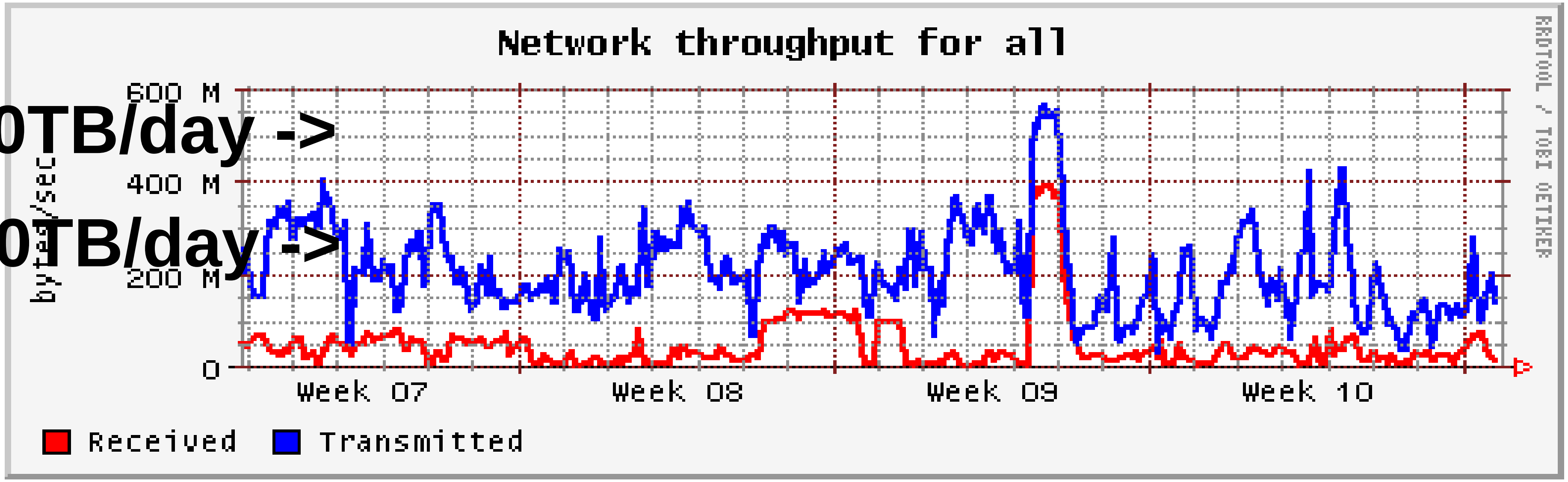
## Hardware Organization:

- Central Analysis Farm (CAF) using FBSNG
- DH using dCache & NFS/rootd
  - ~54TB user scratch (rootd)
  - ~70TB dCache read pools
  - ~26TB NFS/rootd (“legacy”)

# CDF DH Today

## Caching Model for dCache:

- Golden cache: autoload, never delete
- Regular cache: strife for low cache miss rate
- Raw data: essentially a FIFO buffer
- Distinction is driven by physics goals



40TB/day ->

20TB/day ->

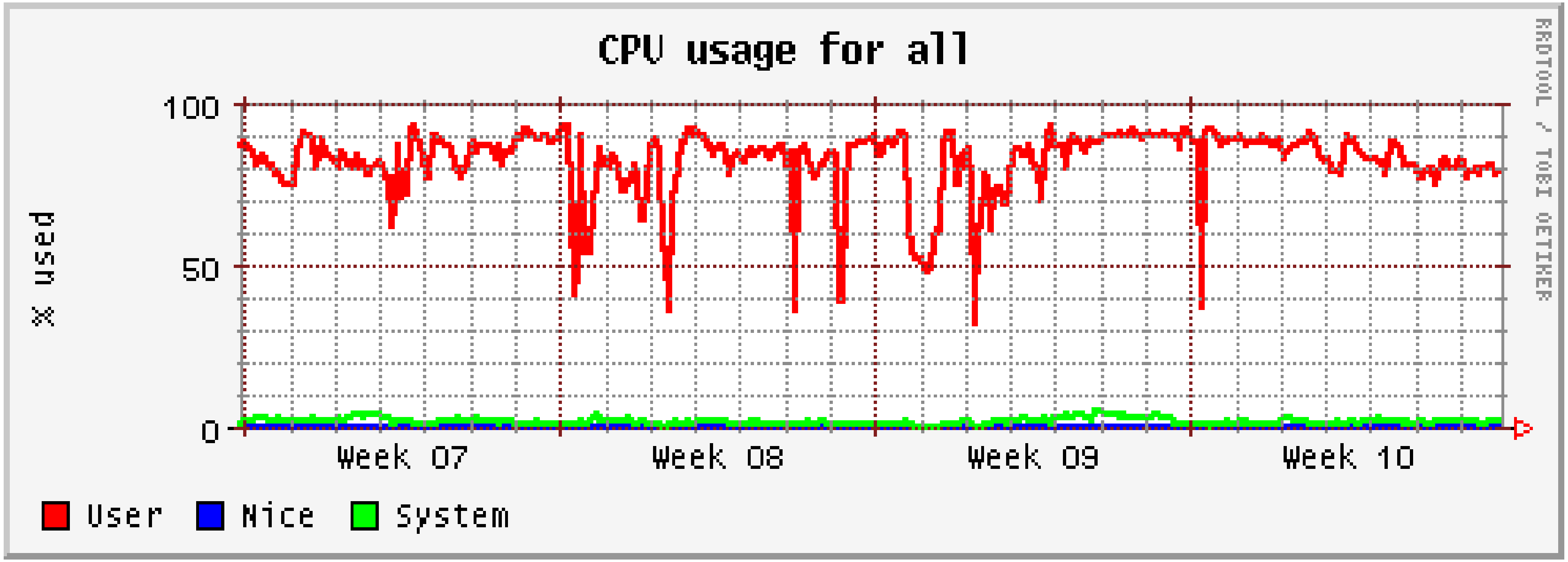
# CAF utilization

## User perspective:

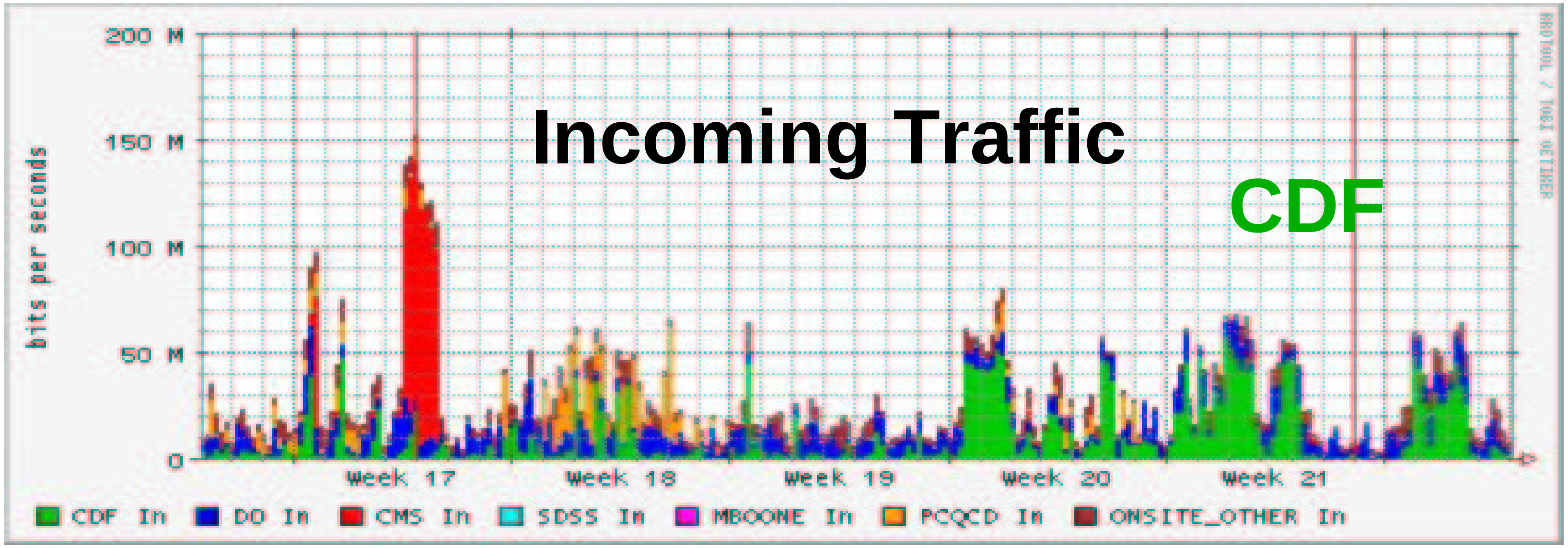
- Up to 10,000 jobs/day
- 400 users total
- 100 users per day

## System perspective:

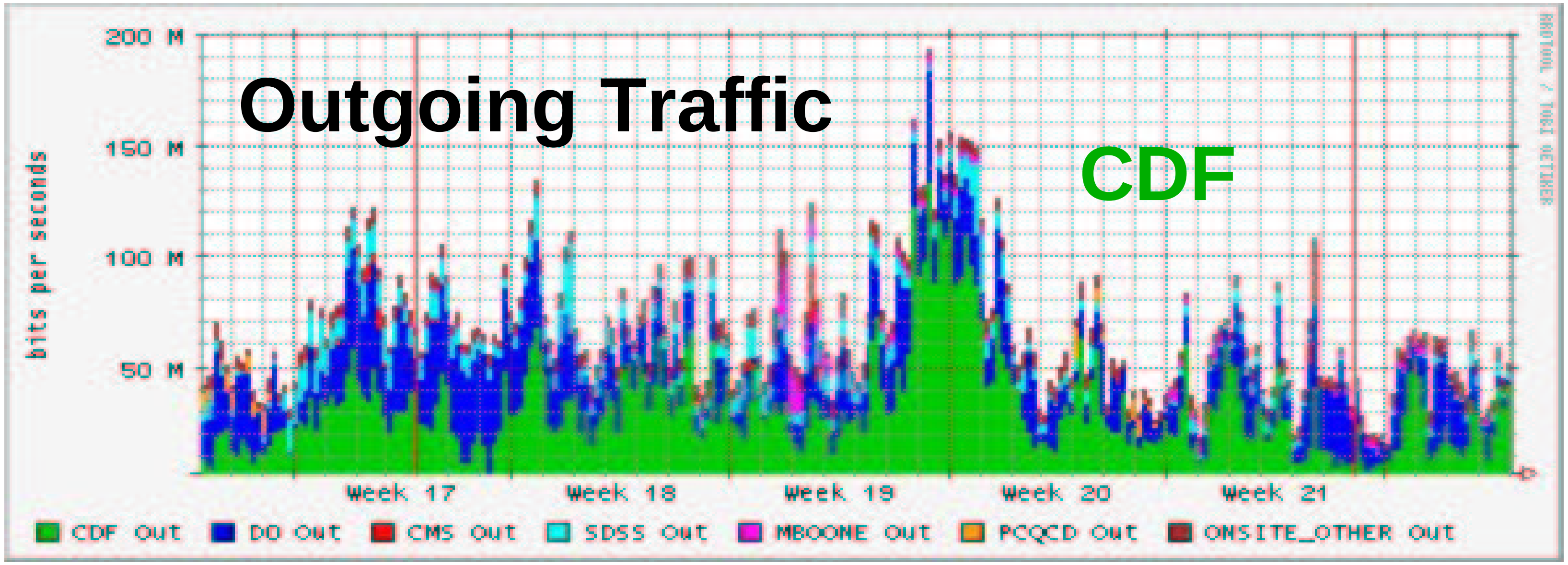
- Up to 90% avg CPU utilization
- 200-600MB/sec I/O
- Failure rate  ~1/2000
- Avg uptime of WN = 60days



# FNAL WAN Activity


Incoming Traffic

CDF


Outgoing Traffic

CDF

# PPDG related activities

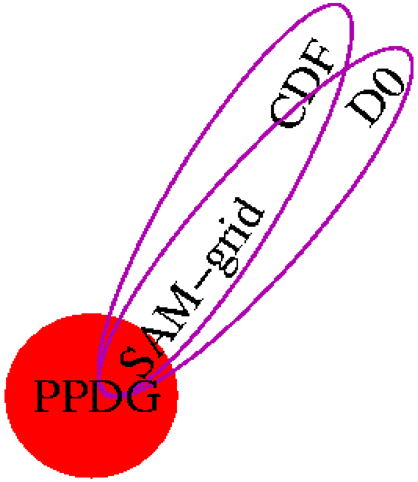
## Goals:

- Better support of offsite computing:
  - MC production (1Million evts/day capacity)
  - User analysis (few small sites, larger sites emerging)
- Co-scheduling of CPU and disk cache @ FNAL
- Better analysis tools support

## Present PPDG related activities in CDF:

- SAM-Grid: D0/CD/CDF joint project
  - SC2002: first physics analysis on sam-grid
- SRM
  - SRM interface to dCache/Enstore to be used by SAM

# SAM-grid @ CDF



- Continued deployment of v1
  - Stability & scalability testing
- Development of v2 functionality
  - Co-scheduling of CPU & data (based on Condor)
  - 'VO management'
  - Improved user interfaces & monitoring
- SRM deployment
  - Sam-dCache integration
  - Stability & scalability testing
  - Implement policies for user write access

**SAM-grid = future of CDF computing**

# 'Long term' Issues

- Need Improved analysis tools support:
  - Prod. software env: ~few Hz max
  - Root 'ntuple': ~few 1000 Hz max
    - Interactive Grid Proposal
- GridPP related activities:
  - Distributed DB project
- Inter grid operability
  - Teragrid: Interactive Grid Proposal
  - Atlas/CMS: Idle non-US resources

# Distributed DB Project

## Implement DB as an abstract concept

- multiple DB types
- freeware slave DB
- Configurable update, incl. Slave triggered
- Use existing grid tools
- Transparent Client -> slave DB connection
- GUI based replication admin tool

**1FTE requested via PPARC e-science**

**interested in collaboration**

contact: Rick St.Denis stdenis@fnal.gov

# IntGrid Vision

- Multi-experiment (BaBar,CDF,CMS,D0, ...)
- Based on common analysis tool: root
- Based on Condor, Globus, SRM
- Build on activities out of CS-2,4,9,11
- Include Non-HEP site: Teragrid
- 2-3 year 2 FTE effort within ppdg
- production quality
  system for Summer Conf. each year.

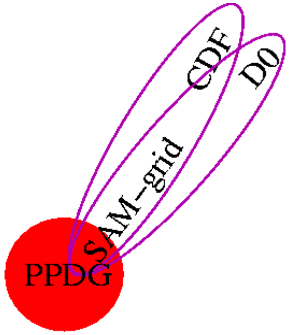
**Use HEP user community in R&D for general int. Grid principles**

# IntGrid Functionality

- User/client perspective:
  - Session start/data decl.:  1-2min
  - Simple query: ~10sec; ~10-20% duty cycle
  - 10-100 'slaves' per user/client
    - Sanity check: 1e7evts * 10kB /(100slaves *10s) = 1Gb/sec
  - up/down load of data & libs fro/to user
  - Automatic log on client node

- 'System' perspective:
  - Global Resource Management (i.e. All clients)
  - Co-location of 'slaves'  with data -> memory cache
  - Batch co-existence (managed suspend/resume)

# Conclusion



CDF has excellent track record of deploying large distributed computing systems.
Focused (mostly) on fabric issues so far.

Strong commitment to existing collaborative efforts with ppdg via D0/CD/CDF joint projects.
Our focus is clearly on deployment of production systems.

New intGrid proposal that builds upon ppdg developments from CS-2,4,11.